Amendments to the Claims

1. (Currently amended) A method for switching fonts without embedding font switches in the data comprising the steps of:

receiving a character <u>to be effectively</u>[[,]] ~~wherein said character is a modified character in a first base font resource or a character to be~~ added to [[or]] , deleted from, or modified in a ~~said~~ first base font resource;

creating a <u>new</u> font resource that ~~comprises said~~ <u>includes the received</u> character; <u>and</u>

linking said ~~created~~ <u>new</u> font resource to said first base font resource <u>to in effect add the received character to, delete the received character from, or modify an existing character in said first base font resource:</u>

<u>wherein said new font resource and said first base font resource act as if they are a unified font resource.</u> ~~if said character is a character to be added; and~~

~~linking said first base font resource to said created font resource if said character is a character to be modified or deleted.~~

2. (Currently amended) The method as recited in claim 1 further comprising the steps of:

creating an entry in a first table indicating said ~~created~~ <u>new</u> font resource is a second base font resource; and

creating a second and a third table associated with said ~~created~~ <u>new</u> font resource, wherein said second table maps code points to glyph indexes, wherein said third table comprises glyphs<u>.</u>

3. (Currently amended) The method as recited in claim 2 further comprising the steps of:

creating a link list in an entry in said first table associated with said first base font resource to link said ~~created~~ <u>new</u> font resource to said first base font resource;

indicating in said entry in said first table associated with said first base font resource to not reverse linking of said first base font resource to said ~~created~~ <u>new</u> font resource if ~~said~~ <u>the received</u> character is a character to be added; and

indicating in said entry in said first table associated with said first base font resource to reverse linking of said first base font resource to said ~~created~~ <u>new</u> font resource if ~~said~~ <u>the received</u> character is a character to be modified or deleted.

4. (Original) The method as recited in claim 3 further comprising the steps of:

receiving an identification of a font resource and a code point; and

transmitting said code point to a rasterizer program associated with said identified font resource.

5. (Original) The method as recited in claim 4, wherein said first base font resource is associated with a fourth table and a fifth table, wherein said fourth table maps code points to glyph indexes, wherein said fifth table comprises glyphs.

6. (Original) The method as recited in claim 5 further comprising the step of:

determining if said code point indexes in said fourth table;

wherein if said code point indexes in said fourth table, then the method further comprises the steps of:

procuring a glyph from said fifth table using a glyph index obtained from said fourth table;

converting said glyph to a bit map representation; and

transmitting said bit map representation to a printer; and

wherein if said code point does not index in said fourth table, then the method further comprises the step of:

determining if said code point indexes in said second table.

7. (Currently amended) The method as recited in claim [[5]] 6 further comprising the step of:

determining if said code point indexes in said second table;

wherein if said code point indexes in said second table, then the method further comprises the steps of:

procuring a glyph from said third table using a glyph index obtained from said second table;

converting said glyph to a bit map representation; and

transmitting said bit map representation to a printer; and

wherein if said code point does not index in said second table, then the method further comprises the step of:

determining if said code point indexes in said fourth table.

8. (Currently amended) A computer program product <u>comprising a computer-readable medium embodying a computer-readable program</u> ~~embodied in a machine readable medium for switching fonts~~ without embedding font switches in the data, <u>wherein the computer-readable program when executed on a computer system causes the computer system to performs the steps of:</u> ~~comprising the programming steps of:~~

receiving a character <u>to be effectively</u> [[,]] ~~wherein said character is a modified character in a first base font resource or a character to be~~ added to [[or]] <u>,</u> deleted from<u>, or modified in a</u> ~~said~~ first base font resource;

creating a <u>new</u> font resource that ~~comprises said~~ <u>includes the received</u> character; <u>and</u>

linking said ~~created~~ <u>new</u> font resource to said first base font resource <u>to in effect add the received character to, delete the received character from, or modify an existing character in said first base font resource;</u>

<u>wherein said new font resource and said first base font resource act as if they are a unified font resource.</u> ~~if said character is a character to be added; and~~

~~linking said first base font resource to said created font resource if said character is a character to be modified or deleted.~~

(Response to 08-21-2007 Office Action)

9. (Currently amended) The computer program product as recited in claim [[10]] 8 further comprising the programming steps of:

creating an entry in a first table indicating said ~~created~~ new font resource is a second base font resource; and

creating a second and a third table associated with said ~~created~~ new font resource, wherein said second table maps code points to glyph indexes, wherein said third table comprises glyphs

10. (Currently amended) The computer program product as recited in claim [[10]] 9 further comprising the programming steps of:

creating a link list in an entry in said first table associated with said first base font resource to link said ~~created~~ new font resource to said first base font resource;

indicating in said entry in said first table associated with said first base font resource to not reverse linking of said first base font resource to said ~~created~~ new font resource if ~~said~~ the received character is a character to be added; and

indicating in said entry in said first table associated with said first base font resource to reverse linking of said first base font resource to said ~~created~~ new font resource if ~~said~~ the received character is a character to be modified or deleted.

11. (Original) The computer program product as recited in claim 10 further comprising the programming steps of:

receiving an identification of a font resource and a code point; and

transmitting said code point to a rasterizer program associated with said identified font resource.

12. (Original) The computer program product as recited in claim 11, wherein said first base font resource is associated with a fourth table and a fifth table, wherein said fourth table maps code points to glyph indexes, wherein said fifth table comprises glyphs.

(Response to 08-21-2007 Office Action)

13. (Original) The computer program product as recited in claim 12 further comprising the programming step of:

    determining if said code point indexes in said fourth table;

    wherein if said code point indexes in said fourth table, then the computer program product further comprises the programming steps of:

        procuring a glyph from said fifth table using a glyph index obtained from said fourth table;

        converting said glyph to a bit map representation; and

        transmitting said bit map representation to a printer; and

    wherein if said code point does not index in said fourth table, then the computer program product further comprises the programming step of:

        determining if said code point indexes in said second table.

14. (Original) The computer program product as recited in claim 13 further comprising the programming step of:

    determining if said code point indexes in said second table;

    wherein if said code point indexes in said second table, then the computer program product further comprises the programming steps of:

        procuring a glyph from said third table using a glyph index obtained from said second table;

        converting said glyph to a bit map representation; and

        transmitting said bit map representation to a printer; and

    wherein if said code point does not index in said second table, then the computer program product further comprises the programming step of:

        determining if said code point indexes in said fourth table.

15. (Currently amended) A system, comprising:

a client configured to generate a first data stream comprising page description information;

a spool coupled to said client, wherein said spool is configured to store said first data stream;

a resource library configured to store a first base font resource;

a print server coupled to said spool and said resource library, wherein said print server comprises:

a first memory unit operable for storing a printer driver configured to generate a second data stream; and

a first processor coupled to said first memory unit; and

a printer coupled to said print server, wherein said printer is configured to receive said second data stream generated from said print server, wherein said printer comprises:

a second memory unit operable for storing a rasterizer program; and

a control unit coupled to said second memory unit;

wherein said client comprises:

a third memory unit operable for storing a computer program for creating a linked resource;

a second processor coupled to said second memory unit, wherein said second processor, responsive to said computer program, comprises:

circuitry operable for receiving a character to be effectively [[,]] ~~wherein said character is a modified character in said first base font resource or a character to be~~ added to, [[or]] deleted from, or modified in a ~~said~~ first base font resource;

circuitry operable for creating a new font resource that ~~comprises said~~ includes the received character; and

circuitry operable for linking said ~~created~~ new font resource to said first base font resource to in effect add the received character to, delete the received character from, or modify an existing character in said first base font resource;

wherein said new font resource and said first base font resource as if they were a unified font resource. ~~if said character is a character to be added; and~~

~~circuitry operable for linking said first base font resource to said created font~~

~~resource if said character is a character to be modified or deleted.~~

16. (Currently amended) The system recited in claim 15, wherein said resource library comprises a first table, wherein said second processor further comprises:

circuitry operable for creating an entry in said first table indicating said ~~created~~ new font resource is a second base font resource; and

circuitry operable for creating a second and a third table associated with said ~~created~~ new font resource to be stored in said resource library, wherein said second table maps code points to glyph indexes, wherein said third table comprises glyphs.

17. (Currently amended) The system as recited in claim 16, wherein said resource library comprises a first table, wherein said second processor further comprises:

circuitry operable for creating a link list in an entry in said first table associated with said first base font resource to link said ~~created~~ new font resource to said first base font resource;

circuitry operable for indicating in said entry in said first table associated with said first base font resource to not reverse linking of said first base font resource to said ~~created~~ new font resource if ~~said~~ the received character is a character to be added; and

circuitry operable for indicating in said entry in said first table associated with said first base font resource to reverse linking of said first base font resource to said ~~created~~ new font resource if ~~said~~ the received character is a character to be modified or deleted.

18. (Original) The system as recited in claim 17, wherein said first processor, responsive to said printer driver, comprises:

circuitry operable for receiving an identification of a font resource and a code point; and

circuitry operable for transmitting said code point to said rasterizer program associated with said identified font resource.

19. (Original) The system as recited in claim 18, wherein said resource library stores a fourth table and a fifth table associated with said first base font resource, wherein said fourth table maps code points to glyph indexes, wherein said fifth table comprises glyphs.

(Response to 08-21-2007 Office Action)

20. (Original) The system as recited in claim 19, wherein said control unit, responsive to said rasterizer program, comprises:

circuitry operable for determining if said code point indexes in said second table;

wherein if said code point indexes in said second table, then said control unit, responsive to said rasterizer program, further comprises:

circuitry operable for procuring a glyph from said third table using a glyph index obtained from said second table;

circuitry operable for converting said glyph to a bit map representation; and

circuitry operable for transmitting said bit map representation to a printer; and

wherein if said code point does not index in said second table, then said control unit, responsive to said rasterizer program, further comprises:

circuitry operable for determining if said code point indexes in said fourth table.

21. (Original) The system as recited in claim 19, wherein said control unit, responsive to said rasterizer program, comprises:

circuitry operable for determining if said code point indexes in said fourth table;

wherein if said code point indexes in said fourth table, then said control unit, responsive to said rasterizer program, further comprises:

circuitry operable for procuring a glyph from said fifth table using a glyph index obtained from said fourth table;

circuitry operable for converting said glyph to a bit map representation; and

circuitry operable for transmitting said bit map representation to a printer; and

wherein if said code point does not index in said fourth table, then said control unit, responsive to said rasterizer program, further comprises:

circuitry operable for determining if said code point indexes in said second table.

22. (Cancelled) A method for switching fonts without embedding font switches in the data comprising the steps of:

    receiving an identification of a font resource;

    searching in the inline data for a native name associated with said font resource; and

    searching a table for said native name associated with said font resource if said native name is not located in said inline data.

23. (Cancelled) The method as recited in claim 22 further comprising the steps of:

    searching in said inline data for a link list associated with said font resource; and

    searching in said table for said link list associated with said font resource if said link list is not located in said inline data;

    searching for a native name associated with any linked font resources identified in said link list in said inline data; and

    searching for said native name associated with any linked font resources identified in said link list in said table if said native name is not located in said inline data.

24. (Cancelled) A computer program product embodied in a machine readable medium for switching fonts without embedding font switches in the data comprising the programming steps of:

    receiving an identification of a font resource;

    searching in the inline data for a native name associated with said font resource; and

    searching a table for said native name associated with said font resource if said native name is not located in said inline data.

25. (Cancelled) The computer program product as recited in claim 24 further comprising the programming steps of:

searching in said inline data for a link list associated with said font resource; and

searching in said table for said link list associated with said font resource if said link list is not located in said inline data;

searching for a native name associated with any linked font resources identified in said link list in said inline data; and

searching for said native name associated with any linked font resources identified in said link list in said table if said native name is not located in said inline data.

26. (Cancelled) A system, comprising:

a client configured to generate a first data stream comprising page description information;

a spool coupled to said client, wherein said spool is configured to store said first data stream;

a resource library configured to store font resources;

a print server coupled to said spool and said resource library, wherein said print server comprises:

a memory unit operable for storing a printer driver configured to generate a second data stream; and

a processor coupled to said first memory unit;

wherein said processor, responsive to said printer driver, comprises:

circuitry operable for receiving an identification of a font resource;

circuitry operable for searching in said first data stream for a native name associated with said font resource; and

circuitry operable for searching a table in said resource library for said native name associated with said font resource if said native name is not located in said first data stream.

27. (Cancelled) The system as recited in claim 26, wherein said processor, responsive to said printer driver, comprises:

circuitry operable for searching in said first data stream for a link list associated with said font resource; and

circuitry operable for searching in said table in said resource library for said link list associated with said font resource if said link list is not located in said first data stream;

circuitry operable for searching for a native name associated with any linked font resources identified in said link list in said first data stream; and

circuitry operable for searching for said native name associated with any linked font resources identified in said link list in said table if said native name is not located in said first data stream.

(Response to 08-21-2007 Office Action)